

AMENDMENTS TO THE CLAIMS

Kindly amend the claims as follows:

1 – 7 (cancelled)

8. (previously presented) A method according to claim 20 and further comprising:
designating any of said objects as “new”; and
deferring the tracing of said “new” objects during any cycle of a plurality of cycles during which any of steps a) – g) are performed.

9. (original) A method according to claim 8 wherein said designating as “new” step is performed if said object is part of an allocation cache from which objects are currently being allocated.

10. (previously presented) A method according to claim 8 and further comprising:
periodically unmarking any marked card containing only “new” objects; and
periodically removing said “new” objects’ “new” designation.

11. (previously presented) A method according to claim 10 wherein said periodically unmarking and periodically removing steps are performed if said object is part of an allocation cache from which objects are not currently being allocated.

12. (currently amended) A method for collecting garbage in a computing environment, the method comprising:

- a) tracing a root object to any of its reachable objects in a population of objects;
- b) marking any of said objects referred to in step a);
- ~~c) unmarking a marked card comprising any of said objects;~~
- d) tracing any of said marked object objects ~~on said unmarked card~~ to an unmarked referent object of said marked object;
- e) marking said unmarked referent object;

- f) tracing said referent object marked in step e) to any of its reachable objects;
- g) marking any of said objects referred to in step f);
- h) tracing any unmarked root object referent to any of its reachable objects;
- i) marking any of said objects referred to in step h);
- j) performing any of steps c) – g); and
- k) designating any unmarked object in said population of objects as available for reallocation,

wherein either of steps a) and f) are performed for a given object only if the card to which the object belongs is not marked, wherein any of steps a) – g) are performed upon said population of objects concurrently with the operation of a mutator upon said population of objects within said computing environment, and wherein any of steps h) – k) are performed upon said population of objects while no mutator operates upon said population of objects within said computing environment.

13. (original) A method according to claim 12 and further comprising marking said card if said mutator modifies an object pointer of an object in said card.

14. (original) A method according to claim 12 wherein any of steps a) – g) are performed concurrently.

15. (original) A method according to claim 12 wherein any of steps h) – j) are performed concurrently.

16 - 19 (cancelled)

20. (currently amended) A method for collecting garbage in a computing environment, the method comprising:

- a) tracing a root object to any of its reachable objects in a population of objects;
- b) marking any of said objects referred to in step a);

~~e) — unmarking a marked card comprising any of said objects;~~
 d) tracing any of said marked ~~object~~ objects ~~on said unmarked card~~ to an unmarked referent object of said marked object;
 e) marking said unmarked referent object;
 f) tracing said referent object marked in step e) to any of its reachable objects;
 g) marking any of said objects referred to in step f);
 h) tracing any unmarked root object referent to any of its reachable objects;
 i) marking any of said objects referred to in step h);
 j) performing any of steps c) – g);
 k) designating any unmarked object in said population of objects as available for reallocation; and
 l) at any time ~~relative to~~ while performing concurrently any of steps a) – g), periodically determining whether a marked card contains at least one of said marked objects, and unmarking any marked card about which it is determined that it does not contain at least one of said marked objects,
 wherein any of steps a) – g) are performed upon said population of objects concurrently with the operation of a mutator upon said population of objects within said computing environment, and wherein any of steps h) – k) are performed upon said population of objects while no mutator operates upon said population of objects within said computing environment.

21. (original) A method according to claim 20 and further comprising marking said card if said mutator modifies an object pointer of an object in said card.

22. (original) A method according to claim 20 wherein any of steps a) – g) are performed concurrently.

23. (original) A method according to claim 20 wherein any of steps h) – j) are performed concurrently.

24 – 37 (cancelled)

38. (previously presented) A system according to claim 44 and further comprising:
means for designating any of said objects as “new”; and
means for deferring the tracing of said “new” objects during any cycle of a plurality of cycles during which any of means a) – g) operate.

39. (original) A system according to claim 38 wherein said means for designating as “new” is operative if said object is part of an allocation cache from which objects are currently being allocated.

40. (original) A system according to claim 38 and further comprising:
means for periodically unmarking any marked card containing only “new” objects; and
means for removing said “new” objects’ “new” designation.

41. (original) A system according to claim 40 wherein said means for periodically unmarking and said means for removing are operative if said object is part of an allocation cache from which objects are not currently being allocated.

42. (currently amended) A system for collecting garbage in a computing environment, the system comprising:

- a) means for tracing a root object to any of its reachable objects in a population of objects;
- b) means for marking any of said objects referred to in a);
- ~~e) means for unmarking a marked card comprising any of said objects;~~
- d) means for tracing any of said marked ~~object~~ objects ~~on said unmarked card~~ to an unmarked referent object of said marked object;
- e) means for marking said unmarked referent object;
- f) means for tracing said marked referent object in e) to any of its reachable objects;
- g) means for marking any of said objects referred to in f);

h) means for tracing any unmarked root object referent to any of its reachable objects;

i) means for marking any of said objects referred to in h); and

j) means for designating any unmarked object in said population of objects as available for reallocation,

wherein either of tracing means a) and f) trace a given object only if the card to which the object belongs is not marked, wherein any of means a) – g) operate upon said population of objects concurrently with the operation of a mutator upon said population of objects within said computing environment, and wherein any of means h) – j) operate upon said population of objects while no mutator operates upon said population of objects within said computing environment.

43. (cancelled)

44. (currently amended) A system for collecting garbage in a computing environment, the system comprising:

a) means for tracing a root object to any of its reachable objects in a population of objects;

b) means for marking any of said objects referred to in a);

~~e) means for unmarking a marked card comprising any of said objects;~~

d) means for tracing any of said marked object objects on said unmarked card to an unmarked referent object of said marked object;

e) means for marking said unmarked referent object;

f) means for tracing said marked referent object in e) to any of its reachable objects;

g) means for marking any of said objects referred to in f);

h) means for tracing any unmarked root object referent to any of its reachable objects;

i) means for marking any of said objects referred to in h);

j) means for designating any unmarked object in said population of objects as available for reallocation; and

k) means for periodically determining at any time concurrently with the operation of any of means a) – g) whether a marked card contains at least one of said marked objects, and unmarking any marked card about which it is determined that it does not contain at least one of said marked objects,

wherein any of means a) – g) operate upon said population of objects concurrently with the operation of a mutator upon said population of objects within said computing environment, and wherein any of means h) – j) operate upon said population of objects while no mutator operates upon said population of objects within said computing environment.

45. (cancelled)

46. (currently amended) A computer program embodied on a computer-readable medium, the computer program comprising:

a) a first code segment operative to trace a root object to any of its reachable objects in a population of objects;

b) a second code segment operative to mark any of said objects referred to in a);

~~e) a third code segment operative to unmark a marked card comprise any of said objects;~~

d) a fourth code segment operative to trace any of said marked ~~object~~ objects on said unmarked card to an unmarked referent object of said marked object;

e) a fifth code segment operative to mark said unmarked referent object;

f) a sixth code segment operative to trace said marked referent object in e) to any of its reachable objects;

g) a seventh code segment operative to mark any of said objects referred to in f);

h) a eighth code segment operative to trace any unmarked root object referent to any of its reachable objects;

i) a ninth code segment operative to mark any of said objects referred to in h); and

j) a tenth code segment operative to designate any unmarked object in said population of objects as available for reallocation,

wherein either of said code segments a) and f) are operative to trace a given object only if the card to which the object belongs is not marked, wherein any of said code segments a) – g) operate upon said population of objects concurrently with the operation of a mutator upon said population of objects within said computing environment, and wherein any of said code segments h) – j) operate upon said population of objects while no mutator operates upon said population of objects within said computing environment.

47. (cancelled)

48. (currently amended) A method for collecting garbage in a computing environment, the method comprising:

a) a first code segment operative to trace a root object to any of its reachable objects in a population of objects;

b) a second code segment operative to mark any of said objects referred to in a);

~~e) a third code segment operative to unmark a marked card comprise any of said objects;~~

d) a fourth code segment operative to trace any of said marked ~~object~~ objects on said unmarked card to an unmarked referent object of said marked object;

e) a fifth code segment operative to mark said unmarked referent object;

f) a sixth code segment operative to trace said marked referent object in e) to any of its reachable objects;

g) a seventh code segment operative to mark any of said objects referred to in f);

h) a eighth code segment operative to trace any unmarked root object referent to any of its reachable objects;

i) a ninth code segment operative to mark any of said objects referred to in h);

j) a tenth code segment operative to designate any unmarked object in said population of objects as available for reallocation; and

k) an eleventh code segment operative ~~prior~~ at any time concurrent to the operation of any of said code segments a) – g), to periodically determine whether a marked card contains at least one of said marked objects, and unmark any marked card about which it is determined that it does not contain at least one of said marked objects,

wherein any of said code segments a) – g) are operative upon said population of objects concurrently with the operation of a mutator upon said population of objects within said computing environment, and wherein any of said code segments h) – j) are operative upon said population of objects while no mutator operates upon said population of objects within said computing environment.

49. (cancelled)